

## REMARKS

Claims 1 and 13 have been amended. No claims have been cancelled or added. Hence, claims 1 – 24 are pending in the Application.

Claims 1 – 24 are presumably rejected under 35 USC 103(a) as being unpatentable over U.S. Patent No. 6,108,653, herein Pereira. The current Office Action states that the Applicant's arguments in the previous response are unpersuasive, states why the arguments are allegedly unpersuasive based on Pereira, and states that claims 1 – 24 are rejected, but does not explicitly state the basis of the rejection. Presumably, the rejections from the previous Office Action are incorporated and maintained in the current Office Action.

## CLAIMS 1 AND 13

Claims 1 and 13 recite:

constructing work granules that manipulate rows in a manner that groups the rows within said work granules according to logical storage units that contain the rows; and during execution by an entity of a particular work granule that involves manipulation operations for rows in a logical storage unit:  
causing said entity to perform said manipulation operations for rows completely contained in said logical storage unit;  
determining that a set of spanning rows that are partially contained in said logical storage unit satisfy a particular condition that relates to which portion of each spanning row of said set of spanning rows resides in said logical storage unit; and  
in response to said determining that a set of spanning rows satisfy a particular condition, causing said entity to perform said manipulation operations for all pieces of all spanning rows in said set of spanning rows.

The system of claims 1 and 13 recite a novel and advantageous approach for modifying data in spanning rows. A spanning row is a row that is stored in more than one logical storage unit, such as a data block. The portions of rows are referred to as row pieces. (See, for example, Application, page 3, lines 8 – 10). The approach claimed requires that an entity, such as a slave process, perform manipulation operations for all pieces of a spanning row when the spanning row satisfies a condition, the condition relating to which portion of the spanning row resides in a particular logical storage unit. The cited art fails to suggest all the features required by claims 1 and 13.

There are at least three features required by claims 1 and 13 that are not suggested much less disclosed by cited art.

**I. Pereira fails to suggest much less disclose constructing work granules that perform operations on rows in a logical storage unit, where those operations are manipulation operations.**

Pereira describes a method for analyzing a table to generate information used by a DBA to maintain a database. The database includes spanning rows in data blocks. Among the information generated is information about the spanning rows, such as the total number of "chained rows", i.e. spanning rows. (col. 8, lines 47 – 49) To gather this information, the task of analyzing the information is distributed to a set of processes or threads, which are assigned a set of blocks to analyze. (Col. 11, lines 38 – 49) The "blocks read are examined to determine the contents of each block. In this manner, block and row statistics are gathered from the blocks read. Once all threads have finished processing their respective assigned set of blocks, summation and averaging of information gathered from each thread is performed (Step 130)." (col. 12, lines 14 – 19)

Apparently, the Office Action has equated constructing work granules to breaking down the task of analyzing data blocks in subtasks which are distributed to the set of processes or threads in Pereira. The subtasks read data from data blocks but do not manipulate the data in the data blocks, e.g. insert or update rows. Thus, at best Pereira teaches constructing subtasks that perform only read operations to data stored in logical storage units, but does not in any way suggest constructing work granules that perform manipulation operations, as claimed.

**II Pereira fails to suggest much less disclose constructing work granules that perform manipulation operations on rows in a logical storage unit, where the logical storage unit stores a spanning row.**

Even though Pereira teaches about spanning rows, it fails to suggest in any way constructing work granules that perform manipulation operations. Therefore, Pereira cannot possibly suggest work granules that perform manipulation operations on logical storage units that include spanning rows, as claimed.

**III Pereira fails to suggest much less disclose determining that set of spanning rows satisfies a particular condition, where the determination is made during the execution of manipulation operations, and where the particular condition relates to which portion of spanning row resides in a particular logical storage unit.**

The Office Action cites col. 4, lines 65 – col. 6, line 13 as disclosing the former limitation of "determining a set of spanning rows that are partially contained in said logical storage unit and that satisfy a particular condition." Presumably, the Office Action is correlating this step to the step of "to determine the information in Table 1". This information includes information about chained rows. Pereira teaches that the information is gathered while reading blocks and generating information about the condition of the database, including the number of chained rows in a table. Thus, assuming Pereira would teach "determining that set of spanning rows ...

satisfy a particular condition", at best, Pereira suggests to perform such a step while performing steps that do not involve manipulation operations, such as when reading blocks and calculating the number of chained rows in a table. Furthermore, while such steps involve the manipulation of information about chained rows, they do not involve or suggest the manipulation of the chained rows themselves, as required by claims 1 and 13. Finally, Pereira does not suggest in any way that a determination of whether to perform a manipulation operation is based in any way on a **condition that relates to which portion of a spanning row resides in a particular logical storage unit.**

As shown above, Pereira fails to suggest much less disclose many features of claims 1 and 13. Therefore, Pereira is patentable. Reconsideration and allowance of claims 1 and 13 is respectfully requested.

#### CLAIMS 8 AND 20

Claims 8 and 20, recite:

inserting a first row piece of a spanning row into a first logical storage unit;  
prior to inserting a second row piece of said spanning row into a second logical storage unit, determining whether one or more criteria is satisfied, wherein said one or more criteria include that said second logical storage unit has enough space allocated to identify at least a threshold number of interested transactions; and  
inserting said second row piece of said spanning row into said second logical storage unit only when said one or more criteria are satisfied.

Claims 8 and 20 describe steps for inserting row pieces of a spanning row in a way that conditions performance of the step on criteria that is based specifically on space needed to

identify a threshold number of interested transactions. Inserting row piece based on criteria that is based on such a factor is not suggested much disclosed by the cited art.

Clearly, the cited art teaches about criteria that controls whether a row is inserted into a data block. These teachings have been alleged to have been taught in col. 3, line 66 – col. 4, line 34, the portion of Pereira cited by the Office Action as disclosing the step of determining whether one or more criteria is satisfied. However, none of the criteria used to govern row insertion taught by Pereira suggests the criteria cited by claims 8 and 20.

Specifically, Pereira teaches that a system wide variable PCTUSED is used to control whether a row is inserted into a particular block. "PCTUSED ... allows the DBA to control the row allocation of a DBMS table. The DBMS prevents additional rows to be placed in a block unless the percentage of that block has fallen below PCTUSED. PCTUSED is different from PCTFREE in the sense that although a block may be used if there is an update, it will not be used to insert a new row unless the percentage used in the block is below PCTUSED. Therefore, PCTFREE is used to maintain free space when the table is initially built, and PCTUSED determines whether a new row may be inserted in a block. The PCTUSED parameter comes into effect when used space becomes free due to deletes." (col. 4, lines 25 – 37)

Thus, Pereira teaches that criteria that controls insertion of a row into a block accounts for whether the total amount of space available in the block is below a threshold. Thus, while Pereira teaches a determination of whether to insert a row is based on criteria and a threshold that accounts for the total space available in the data block, nothing in Pereira suggests criteria or a threshold that accounts for space specifically available to identify a threshold number of transactions, as claimed. In fact, nothing in Pereira teaches that the number of interested transactions that can be tracked by information in a data block is in anyway a factor to be considered for any sort of operation, much less the operation of inserting a row.

An illustration of the difference between the way the system of Pereira inserts rows and a way an embodiment of claim 8 would insert rows is helpful. For purposes of illustration, assume that the database system described Pereira is used to store data. A quarter of the space in the data block is free. PCTUSED and PCTFREE are set to 50% of the free space in the datablock. The datablock contains at least one row piece of a spanning row. The space allocated to identifying interested transactions must be sufficient to identify five transactions. The space allocated in the data block is only sufficient to identify four transactions. Under Pereira, another row piece would be inserted because the total free space available in the block is below either PCTUSED or PCTFREE. Under the embodiment of claim 8, the other row piece is not inserted because there is insufficient space to identify five transactions.

As shown above, Pereira fails to suggest much less disclose many features of claims 8 and 20. Therefore, Pereira is patentable. Reconsideration and allowance of claims 8 and 20 is respectfully requested.

#### DEPENDANT CLAIMS

The pending claims not discussed so far are dependant claims that depend on an independent claim that is discussed above. Because each of the dependant claims include the limitations of claims upon which they depend, the dependant claims are patentable for at least those reasons the claims upon which the dependant claims depend are patentable. Removal of the rejections with respect to the dependant claims and allowance of the dependant claims is respectfully requested. In addition, the dependent claims introduce additional limitations that independently render them patentable. Due to the fundamental difference already identified, a separate discussion of those limitations is not included at this time.

For the reasons set forth above, Applicant respectfully submits that all pending claims are patentable over the art of record, including the art cited but not applied. Accordingly, allowance of all claims is hereby respectfully solicited.

Respectfully submitted,

HICKMAN PALERMO TRUONG & BECKER LLP



Dated: February 3, 2004

Marcel K. Bingham  
Reg. No. 42,327

1600 Willow Street  
San Jose, CA 95125  
Telephone No.: (408) 414-1080 ext.206  
Facsimile No.: (408) 414-1076

**CERTIFICATE OF MAILING**

I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail in an envelope addressed to: Mail Stop AF, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450

on 2/3/04  
(Date)

by Rudy Bingham  
(Signature)